

Цели и задачи

Цель лабораторной работы

Изучение алгоритма шифрования гаммированием

Выполнение лабораторной работы

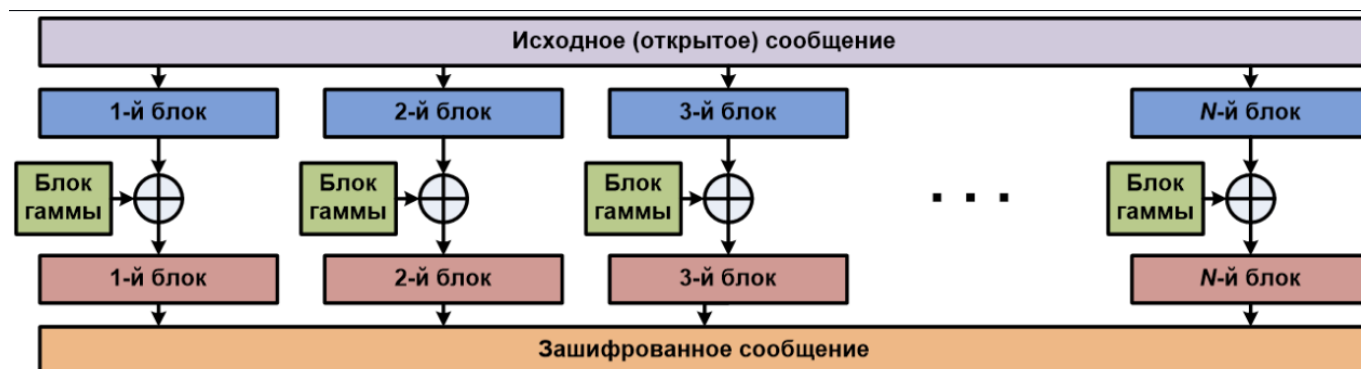
Гаммирование

Гаммирование – это наложение (снятие) на открытые (зашифрованные) данные криптографической гаммы, т.е. последовательности элементов данных, вырабатываемых с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных.

Гаммирование

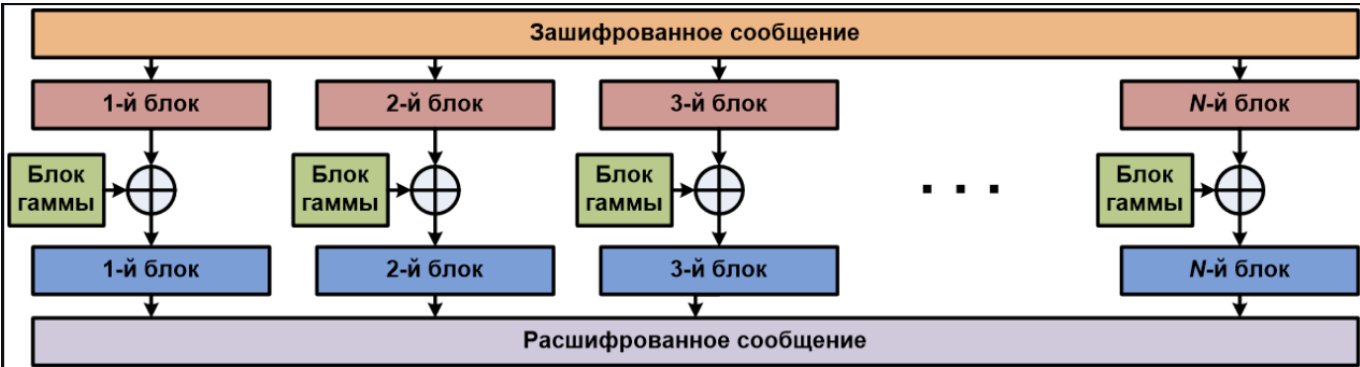
Наложение (или снятие) гаммы на блок сообщения в рассматриваемом нами стандарте реализуется с помощью операции побитного сложения по модулю 2 (XOR). То есть при шифровании сообщений каждый блок открытого сообщения XORится с блоком криптографической гаммы, длина которого должна соответствовать длине блоков открытого сообщения. При этом, если размер блока исходного текста меньше, чем размер блока гаммы, блок гаммы обрезаается до размера блока исходного текста (выполняется процедура усечения гаммы).

Алгоритм



{ #fig:001 }

Алгоритм



{ #fig:002 }

Формула

В аддитивных шифрах символы исходного сообщения заменяются числами, которые складываются по модулю с числами гаммы. Ключом шифра является гамма, символы которой последовательно повторяются. Перед шифрованием символы сообщения и гаммы заменяются их номерами в алфавите и само кодирование выполняется по формуле

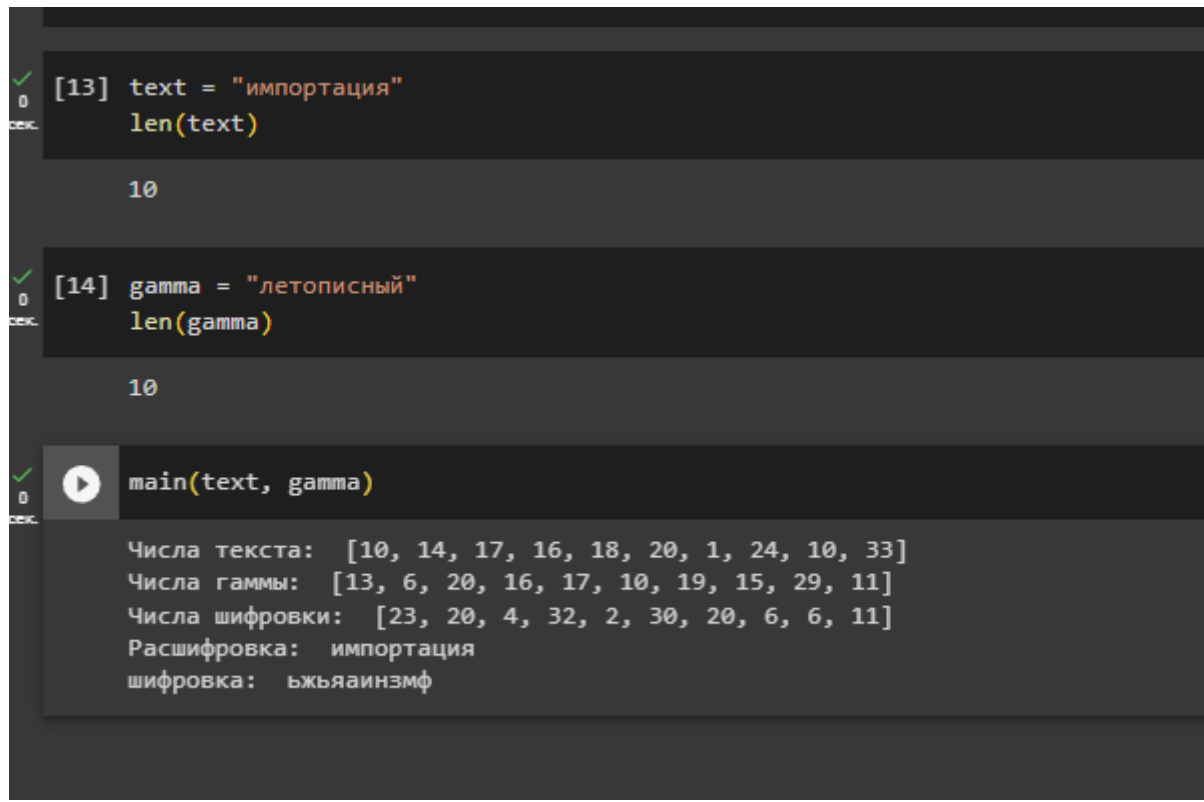
$$C_i = (T_i + G_i) \bmod N$$

Пример работы алгоритма

<i>T</i>	К	А	Ф	Е	Д	Р	А		С	И	С	Т	Е	М		И	Н	Ф	О	Р	М	А	Т	И	К	И
<i>G</i>	С	И	М	В	О	Л	С	И	М	В	О	Л	С	И	М	В	О	Л	С	И	М	В	О	Л	С	И
<i>T</i>	12	1	22	6	5	18	1	34	19	10	19	20	6	14	34	10	15	22	16	18	14	1	20	10	12	10
<i>G</i>	19	10	14	3	16	13	19	10	14	3	16	13	19	10	14	3	16	13	19	10	14	3	16	13	19	10
<i>T+G</i>	31	11	36	9	21	31	20	44	33	13	35	33	25	24	48	13	31	35	35	28	28	4	36	23	31	20
<i>mod N</i>	31	11	36	9	21	31	20	0	33	13	35	33	25	24	4	13	31	35	35	28	28	4	36	23	31	20
<i>0 → N</i>	31	11	36	9	21	31	20	44	33	13	35	33	25	24	4	13	31	35	35	28	28	4	36	23	31	20
<i>C</i>	Э	Й	1	З	У	Э	Т	9	Я	Л	О	Я	Ч	Ц	Г	Л	Э	О	О	Ъ	Ъ	Г	1	Х	Э	Т

{ #fig:003 }

Пример работы программы



The screenshot shows a Jupyter Notebook interface with three code cells. The first cell defines a variable 'text' as 'импортация' and prints its length, which is 10. The second cell defines a variable 'gamma' as 'летописный' and prints its length, which is 10. The third cell calls a function 'main(text, gamma)' which outputs several arrays of numbers and the original text and its encrypted version.

```
[13] text = "импортация"
len(text)

10

[14] gamma = "летописный"
len(gamma)

10

main(text, gamma)

Числа текста: [10, 14, 17, 16, 18, 20, 1, 24, 10, 33]
Числа гаммы: [13, 6, 20, 16, 17, 10, 19, 15, 29, 11]
Числа шифровки: [23, 20, 4, 32, 2, 30, 20, 6, 6, 11]
Расшифровка: импортация
шифровка: ьжъяаинзмф
```

{ #fig:004 }

Выводы

Результаты выполнения лабораторной работы

Изучили алгоритм шифрования с помощью гаммирования